



Tecnológico Nacional De México Instituto Tecnológico De Milpa Alta II



Ingeniería en Sistemas Computacionales

Asignatura:
Mean Stack Para Front-end

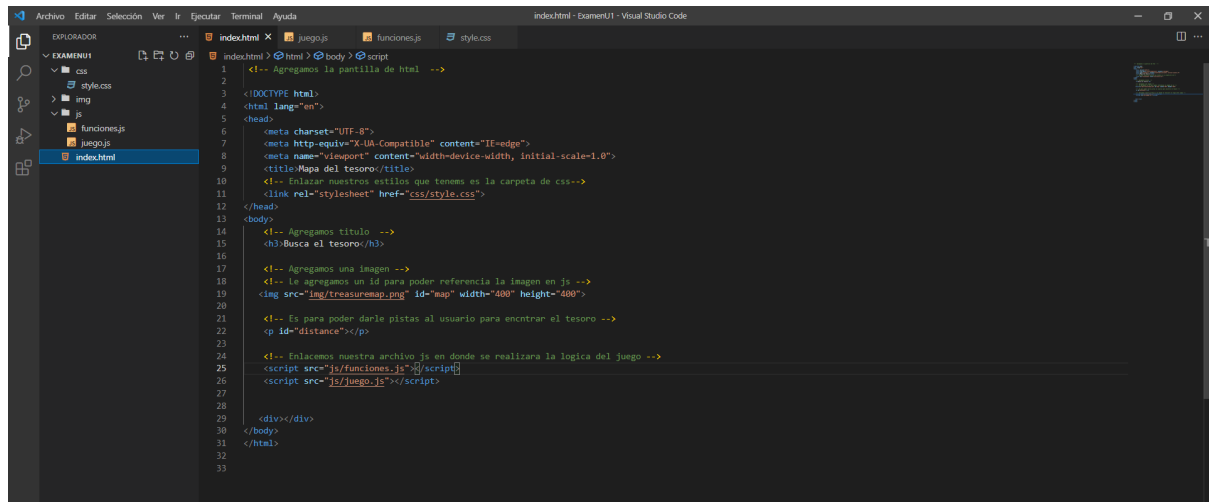
Unidad 1
Examen “Juego en Javascript”

Estudiante:
Martínez Zaragoza Laura Itzel

Profesor:
Roldan Aquino Segura

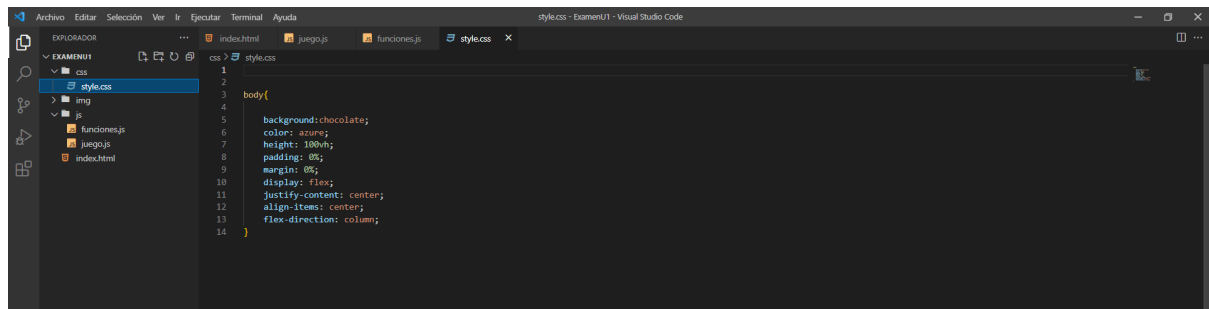
30-septiembre-2021

1.-Lo primero que se realizó fue crear la plantilla de html en donde colocaremos el título del juego, la imagen, el estilo de la pagina que es el css y en donde se va mostrar las pistas al usuario de igual manera el archivo de js en donde se realizará en funcionamiento del juego.



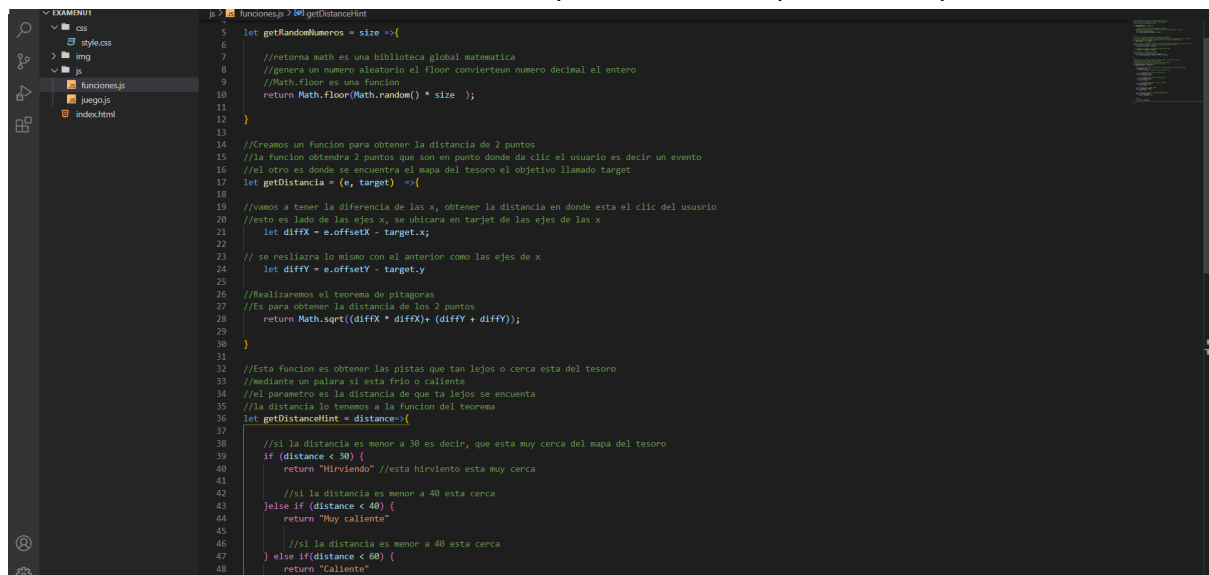
```
1 <!-- Agregamos la plantilla de html -->
2
3 <!DOCTYPE html>
4 <html lang="en">
5 <head>
6   <meta charset="UTF-8">
7   <meta http-equiv="X-UA-Compatible" content="IE=edge">
8   <meta name="viewport" content="width=device-width, initial-scale=1.0">
9   <title>Mapa del tesoro</title>
10  <!-- Enlazamos nuestros estilos que tenemos en la carpeta de css -->
11  <link rel="stylesheet" href="css/style.css">
12 </head>
13 <body>
14  <!-- Agregamos titulo -->
15  <h3>Busca el tesoro</h3>
16
17  <!-- Agregamos una imagen -->
18  <!-- Le agregamos un id para poder referencia la imagen en js -->
19  
20
21  <!-- Es para poder darle pistas al usuario para encontrar el tesoro -->
22  <p id="distance"></p>
23
24  <!-- Enlazamos nuestro archivo js en donde se realizara la logica del juego -->
25  <script src="js/funciones.js"></script>
26  <script src="js/juego.js"></script>
27
28
29 </body></div>
30 </body>
31 </html>
32
33
```

2.-De igual manera realizamos los estilos que tendrá nuestra pagina en la web que el color del body, centrado de la imagen y texto.



```
1
2
3 body{
4
5   background:chocolate;
6   color: brown;
7   height: 1000px;
8   padding: 0%;
9   margin: 0%;
10  display: flex;
11  justify-content: center;
12  align-items: center;
13  flex-direction: column;
14 }
```

3.-Creamos un archivo llamado funciones.js es en donde se realiza la lógica de juego, en donde crearemos las funciones para cada operación que se necesite.



```
1 //funciones.js
2
3 //funciones.js
4
5 let getRandomNumeros = size =>{
6
7   //retorna math es una biblioteca global matematica
8   //genera un numero aleatorio el floor convierteun numero decimal el entero
9   //Math.floor es una funcion
10  return Math.floor(Math.random() * size );
11 }
12
13
14 //Creamos un funcion para obtener la distancia de 2 puntos
15 //la funcion obtendra 2 puntos que son en punto donde da clic el usuario es decir un evento
16 //el otro es donde se encuentra el mapa del tesoro el objetivo llamado target
17 let getDistancia = (e, target) =>{
18
19   //Vamos a tener la diferencia de las x, obtener la distancia en donde esta el clic del usuario
20   //esto es lado de las ejes x, se ubicara en target de las ejes de las x
21   let diffX = e.offsetX - target.x;
22
23   // se repitiera lo mismo con el anterior como las ejes de x
24   let diffY = e.offsetY - target.y
25
26   //Realizaremos el teorema de pitagoras
27   //Es para obtener la distancia de dos puntos
28   return Math.sqrt((diffX * diffX) + (diffY * diffY));
29 }
30
31
32 //Esta funcion es obtener las pistas que tan lejos o cerca esta del tesoro
33 //mediante un palabra si esta frio o caliente
34 //el parametro es la distancia de que ta lejos se encuentra
35 //la distancia lo tenemos a la funcion del teorema
36 let getDistanciaHint = distance=>{
37
38   //si la distancia es menor a 30 es decir, que esta muy cerca del mapa del tesoro
39   if (distance < 30) {
40     return "Hirviendo" //esta hirviendo esta muy cerca
41   }
42   //si la distancia es menor a 40 esta cerca
43   }else if (distance < 40) {
44     return "Muy caliente"
45   }
46   //si la distancia es menor a 60 esta cerca
47   } else if (distance < 60) {
48     return "Caliente"
49   }
50 }
```

4.- Se realizó en la primera función crear los números aleatorios con las medidas de ancho y largo de la imagen para tener un límite, en la cual fueron las coordenadas del tesoro.

```
js > funciones.js > ...
1
2 //Esta función se encarga de generar numeros aleatorios
3 //Esta función fue creada a la versión 6 de javascript
4 //Es una función de parámetro flecha
5
6 let getRandomNumeros = size =>{
7
8     //retorna math es una biblioteca global matemática
9     //genera un número aleatorio el floor convierte un número decimal en entero
10    //Math.floor es una función
11    return Math.floor(Math.random() * size );
12
13 }
14
15 //Creamos una función para obtener la distancia de 2 puntos
16 //la función obtendrá 2 puntos que son en punto donde da clic el usuario es decir un evento
17 //el otro es donde se encuentra el mapa del tesoro el objetivo llamado target
18 let getDistancia = (e, target) =>{
19
20    //vamos a tener la diferencia de las x, obtener la distancia en donde está el clic del usuario
21    //esto es lado de las ejes x, se ubicará en target de las ejes de las x
22    let diffX = e.offsetX - target.x;
23
24    //se realizó lo mismo con el anterior como las ejes de y
25    //esta parte es con las ejes de y
26    let diffY = e.offsetY - target.y
27
28    //Realizaremos el teorema de pitágoras
29    //Es para obtener la distancia de los 2 puntos
30    return Math.sqrt((diffX * diffX) + (diffY * diffY));
31
32 }
```

5.- En la segunda función se realizó para obtener la distancia de los puntos de los ejes x y los ejes Y, mediante la operación del teorema de Pitágoras mediante el clic del mouse calcular el punto.

```
js > funciones.js > ...
13
14 }
15
16 //Creamos una función para obtener la distancia de 2 puntos
17 //la función obtendrá 2 puntos que son en punto donde da clic el usuario es decir un evento
18 //el otro es donde se encuentra el mapa del tesoro el objetivo llamado target
19 let getDistancia = (e, target) =>{
20
21    //vamos a tener la diferencia de las x, obtener la distancia en donde está el clic del usuario
22    //esto es lado de las ejes x, se ubicará en target de las ejes de las x
23    let diffX = e.offsetX - target.x;
24
25    //se realizó lo mismo con el anterior como las ejes de y
26    //esta parte es con las ejes de y
27    let diffY = e.offsetY - target.y
28
29    //Realizaremos el teorema de pitágoras
30    //Es para obtener la distancia de los 2 puntos
31    return Math.sqrt((diffX * diffX) + (diffY * diffY));
32
33 }
```

6.-Se realizó la tercera función en donde el usuario al hacer clic encontrará pistas si está cerca o lejos mediante la distancia, en la cual se utilizó la estructura de decisión.

```

js > funciones.js > ...
34 //Esta función es obtener las pistas que tan lejos o cerca esta del tesoro
35 //mediante un palabra si esta frio o caliente
36 //el parametro es la distancia de que ta lejos se encuentra
37 //la distancia lo tenemos a la función del teorema
38 let getDistanceHint = distance=>{
39
40     //si la distancia es menor a 30 es decir, que esta muy cerca del mapa del tesoro
41     if (distance < 30) {
42         return "Hirviendo" //esta hirviendo esta muy cerca
43     }
44     //si la distancia es menor a 40 esta cerca
45     }else if (distance < 40) {
46         return "Muy caliente"
47     }
48     //si la distancia es menor a 40 esta cerca
49     } else if(distance < 60) {
50         return "Caliente"
51     }
52     //si la distancia es menor a 100 es caliente
53     }else if(distance < 100){
54         return "Calido"
55     }
56     //si la distancia es menor a 180
57     }else if(distance < 180){
58         return "frio"
59     }
60     //si la distancia es menor a 360 esta demasiado frio
61     }else if(distance<360){
62         return "realmente frio"
63     }
64     }else {
65         return "Congelado"
66     }
67 }

```

7.-Creamos otro archivo llamado juego.js en donde se llamará las funciones creadas en la cual solo para poder visual en la consola en donde se encuentra las coordenadas del tesoro, también la distancia que se encuentra el usuario a dar clic en cualquier parte de la imagen y por último la pista que si está cerca o lejos.

```

js > juego.js > target
16 //se guarda en la variable $map y hace referencia a un elemento de html
17 let $map = document.getElementById('map');
18
19 //lo guardamos en una variable en la cual esta referencia en una etiqueta de html
20 let $distance = document.getElementById('distance');
21
22 //obtenido lo click van a ir incrementado
23 let clicks = 0;
24
25
26 // se agrega un escuchador para el evento clic
27 //es cuando el usuario hace clic se va realizar algo
28 $map.addEventListener('click', function (e){
29
30     //un contador de clicks que va incrementando
31     clicks++;
32
33     //obtener la coordena del usuario
34     //en la variable de distance se utilizara la funcion en donde se realizo el torema de pista
35     let distance = getDistancia(e,target);
36
37     //se mostrara los datos es decir la dñcia en donde se encuentra en usuario
38     console.log(distance);
39
40     //esto nos volvera las palabra que se encuentra frio o caliente
41     //lo guardamos en una variable
42     let distanceHint = getDistanceHint(distance);
43
44     //es para mostran en consola
45     console.log(distanceHint );
46
47     //la etiqueta colcaremos el string que nos esta devolviendp
48     $distance.innerHTML = `<h1>${distanceHint}</h1>`;
49
50     //si la distancia es menor a 20 esta bastante cerca
51
52     if (distance < 20) {
53         alert('Haz encontrado el tesoro :D en ${clicks} clicks!');
54
55         //refrescar la pagina con javascript
56         location.reload();
57     } else {
58
59
60     }
61 });

```

http://127.0.0.1:5500

DevTools is now available in Spanish! [Always match Chrome's language](#) [Switch DevTools to Spanish](#)

Don't show again

Elements Console Sources Network

Filter Default levels

No issues

Hide network ☐ Preserve log ☐ Selected context only ☐ Group similar messages in console ☒ Log XMLHttpRequests ☐ Eager evaluation ☒ Autocomplete from history ☒ Evaluate triggers user activation

This sidebar will be removed in a future version of Chrome. If you have feedback, please let us know via the [issue tracker](#).

3 messages

- {x: 20, y: 34} juego.js:12
- 201.97277044195835 juego.js:38
- realmente frio juego.js:45

3 user mess...

No errors

No warnings

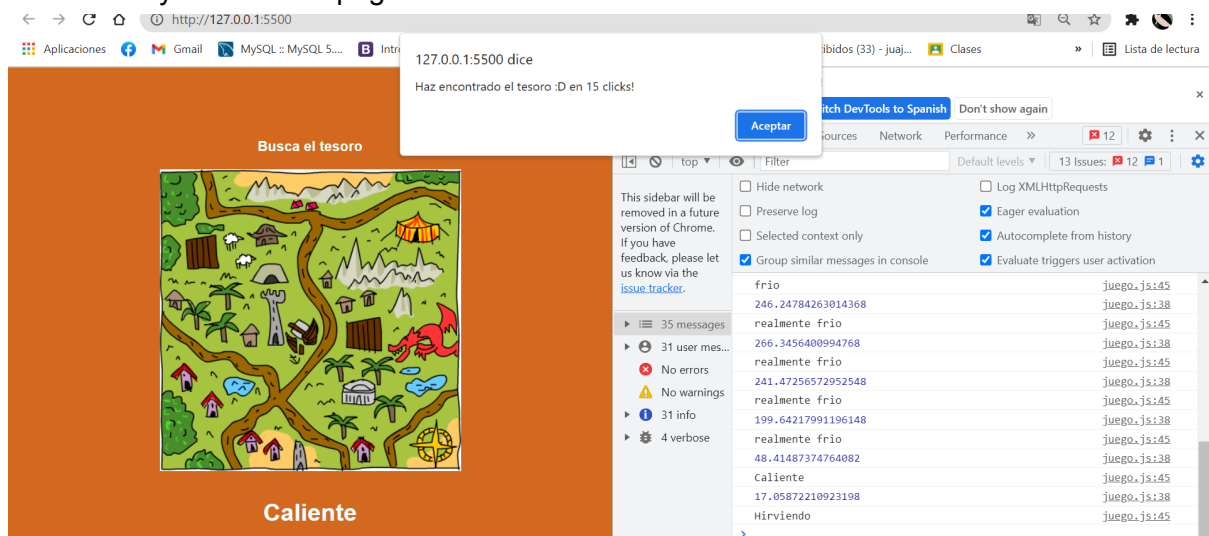
3 info

No verbose

Console What's New Issues

Highlights from the Chrome 94 update

8.-Al final se agregó un alert por si encuentra el tesoro y cuantos clics realizó para encontrarlo y refrescar la página al encontrar en tesoro.



Funcionamiento del juego

9.-Jugamos el juego para encontrar el tesoro en la cual nos daban las pistas que estábamos caliente del tesoro



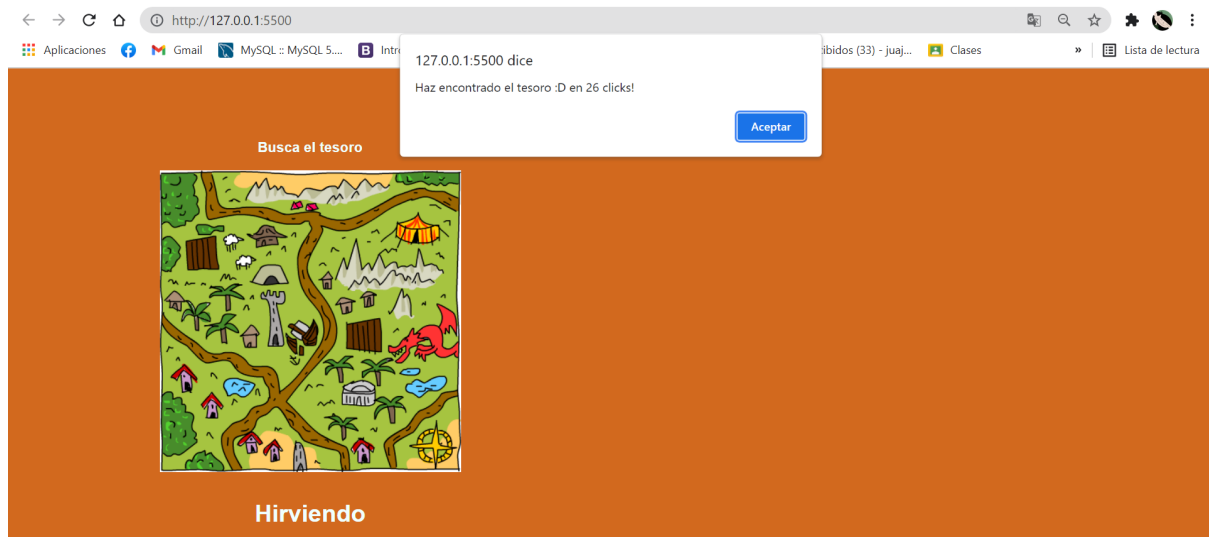
10.-Lo intentamos realizamos otro clic nos dio la pista que estábamos más cerca del tesoro



11.-Volvimos a intentar y nos alejamos del tesoro y estábamos realmente fríos.



12.-Hasta que al final encontramos el tesoro realizamos 26 clics para encontrar el tesoro y nos aviso mediante el alert que agregamos.



Conclusión: Lo que se aprendió fue en como poner el práctica el manejo de funciones de otra version mas actual que fue la funcion de fecha, de igual cómo poder calcular las distancias de los puntos utilizando la biblioteca de Math y como pasarlos de decimales a enteros, realizar la manipulación de una imagen de mapas de tan solo dando clic puedes encontrar las distancias y al final crear un juego en javascript usando programación sencilla y poder ejecutar sin ningún problema.